Journal of Information Processing Vol.28 889–902 (Dec. 2020)

[DOI: 10.2197/ipsjjip.28.889]

Regular Paper

Gossiping with Interference Constraints in Radio Chain Networks

Jean-Claude Bermond[1,a)] Takako Kodate[2,b)] Joseph Yu[3,c)]

Received: December 31, 2019, Accepted: September 10, 2020

Abstract: In this paper, we study the problem of gossiping with neighboring interference constraint in radio chain networks. Gossiping (also called total exchange information) is a protocol where each node in the network has a message and is expected to distribute its own message to every other node in the network. The gossiping problem consists in finding the minimum running time (makespan) of a gossiping protocol and efficient algorithms that attain this makespan. We focus on the case where the transmission network is a chain (directed path or line) network. We consider synchronous protocols where it takes one unit of time (step) to transmit a unit-length message. During one step, a node receives at most one message only through one of its two neighbors. We assume that during one step, a node cannot be both a sender and a receiver (half duplex model). Moreover we have neighboring interference constraints under which a node cannot receive a message if one of its neighbors is sending. A round consists of a set of non-interfering (or compatible) calls and uses one step. We completely solve the gossiping problem for P_n, the chain network on n nodes, and give an algorithm that completes the gossiping in $3n - 5$ rounds (for $n > 3$), which is exactly the makespan.

Keywords: gossiping, radio networks, interference, chains, paths

1. Introduction and Notations

The aim of this paper is to design optimal gossiping (or total exchange information) protocols for chain networks with neighboring interference constraints. This paper answers a problem considered in Ref. [6] where readers can find further references and motives. More precisely our transmission network is a chain (also called line) modeled by a symmetric dipath P_n (symmetric signifies that the communication here is bidirectional). The nodes are labeled from 0 to $n - 1$, and each node i has a message also denoted i. The arcs represent the possible communications. They are in the form $(i, i+1)$, $0 \le i \le n-2$ and $(i, i-1)$, $1 \le i \le n-1$. In a gossiping protocol, each node wants to distribute its own message to every other node in the network. The network is assumed to be synchronous and the time is slotted into *steps*. During a step, a node receives at most one message only through one of its neighbors. One important feature of our model is the assumption that a node can either transmit or receive at most one message per step. In particular, we do not allow concatenation of messages.

We will consider only useful (valid) calls in which the sender sends a message to a receiver only if it is unknown to the receiver. We can have two types of sendings as follows:

(a) via a (regular) call $(i, i+1)$ (resp. $(i, i-1)$), in which the node i sends to the right (resp. to the left) i.e. to the node $i + 1$ (resp. $i - 1$) one message which is not known to the node $i + 1$ (resp. $i - 1$)

(b) via a 2-call $\{i : i - 1, i + 1\}$, in which the node i sends at the same time to both nodes $i - 1$ and $i + 1$ one message which is not known to both nodes and so the message must be i.

We assume that each device is equipped with a half duplex interface, i.e., a node can receive or send, but cannot both receive and send during a step. Furthermore we use a primary node interference model like the one used in Refs. [1], [2], [3], [6], [7]. In this model, when one node is transmitting, its own power prevents any other signal from being properly received by its neighbors (near-far effect of antennas). So two calls (s, r) and (s', r') interfere if the distance between s and r', $d(s, r') \le 1$ or the distance between s' and r, $d(s', r) \le 1$. For example call $(i, i+1)$ will interfere with all the following calls $(i-2, i-1)$, $(i-1, i)$, $(i+1, i)$, $(i+1, i+2)$, $(i+2, i+1)$ and $(i+2, i+3)$. Two non-interfering calls will be called compatible. Therefore the two calls (s, r) and (s', r') are compatible if $d(s, r') > 1$ and $d(s', r) > 1$. For example call $(i, i+1)$ is compatible with calls $(i-1, i-2)$ and $(i+3, i+2)$. Only non-interfering (or compatible) calls can be performed in the same step and we will define a round as a set of compatible calls.

The gossiping problem consists in finding the minimum running time (makespan) of a gossiping protocol, i.e., the minimum number R_n of rounds needed to complete the gossiping and to find efficient algorithms that attain this makespan.

On problems related to information dissemination, we refer to the survey in Ref. [4]. The gossiping problem has been studied in both full duplex and half duplex models (i.e., without interfer-

1 Université Côte d'Azur, CNRS, Inria, I3S, France
2 Department of Information and Sciences,Tokyo Woman's Christian University, Japan
3 Department of Mathematics, University of the Fraser Valley, B.C., Canada
a) jean-claude.bermond@inria.fr
b) kodate@lab.twcu.ac.jp
c) joseph.yu@ufv.ca

ences) with no bounds on the message size. A survey for gossiping with the interference model considered in this paper has been completed as in Ref. [5] but most of the results concern unbounded message sizes and concatenation is allowed.

The gossiping problem with unit length messages and neighboring interference (our model) was first studied in Ref. [6]. The authors established that the makespans of gossiping protocols in chain (called line) and ring networks with n nodes are $3n + \Theta(1)$ and $2n + \Theta(1)$ respectively. They gave for general graphs an upper bound of $O(nlog^2 n)$. This bound was improved in Ref. [8] to $O(nlogn)$ with the help of a probabilistic argument. In Ref. [3], we completely solved the gossiping problem in radio ring networks with the same model (our results depend on the congruence of n modulo 12).

Furthermore in Ref. [6], the authors proved for the chain P_n a lower bound of $3n-6$ and gave a sophisticated protocol in $3n+12$ rounds. Here we determine exactly the minimum number R_n of rounds needed to complete the gossiping when the transmission network is a chain P_n on n nodes based on the model described above (see Theorem 1). To our best knowledge no improvement were made to these bounds since 2002 and the determination of the exact value of R_n remains an unsolved problem. When we tackled the problem some years ago, we quickly found a better lower bound of $3n - 5$ when $n \geq 4$. We found also optimal protocols meeting this bound for small values of n, but were unable to give a general protocol that works for all n. We were more successful with rings, but the tools developed in Ref. [3] for rings cannot be used for chains. Surprisingly the problem for chains appears to be very complicated due to the bottleneck in the middle of the chain. Eventually, we succeeded in designing an optimal protocol by developing new sophisticated tools.

Theorem 1 *The minimum number R_n of rounds needed to complete the gossiping in the chain network P_n ($n \geq 3$) with the neighboring interference model and unit length messages is*

$$R_n = \begin{cases} 3n-5 & \text{if } n \geq 4 \\ 5 & \text{if } n = 3. \end{cases}$$

Remark that for $n = 3$, Theorem 1 can be proven easily. We have 6 calls to perform, and a round contains one call except for the unique round containing the 2-call $\{1 : 0, 2\}$. Therefore at least 5 rounds are needed. The following five calls will work: $\{1 : 0, 2\}$ with message 1, $(0, 1)$ and $(1, 2)$ with message 0, $(2, 1)$ and $(1, 0)$ with message 2.

In the remainder of the paper we assume that $n \geq 4$.

2. Lower Bound for $n \geq 4$

Proposition 1 *The minimum number R_n of rounds needed to complete the gossiping in the chain network P_n ($n \geq 4$), with the neighboring interference model and unit length messages satisfies $R_n \geq 3n - 5$.*

Proof. We first remark that for a given i, $i+1$ messages should be transmitted via the call $(i, i+1)$ (namely the messages $0 \leq j \leq i$) and $n-i-1$ messages via the call $(i+1, i)$ (namely the messages $i+1 \leq j \leq n-1$). So altogether, calls $(i, i+1)$ and $(i+1, i)$ are used to transmit n messages. Consider $n \geq 4$. From the remark, in order to complete any gossiping scheme, there are $3n$ messages that are needed to be transmitted using one of the six calls $(0, 1)$, $(1, 0)$, $(1, 2)$, $(2, 1)$, $(2, 3)$ and $(3, 2)$: the message 0 over $(0, 1)$, the messages $1, 2, \ldots, n-1$, over $(1, 0)$, and similarly, two messages over $(1, 2)$, $n-2$ messages over $(2, 1)$, three messages over $(2, 3)$ and $n-3$ messages over $(3, 2)$, or $3n$ messages in total. Now we will count the number of rounds needed to transmit the $3n$ messages using one of the 6 calls $(0, 1)$, $(1, 0)$, $(1, 2)$, $(2, 1)$, $(2, 3)$, and $(3, 2)$. In general, a round contains at most one of these 6 calls, except if the round is of the following four types in which case it can contain two calls.

- type 1: a round with the unique 2-call $\{1 : 0, 2\}$. Note that there is at most one such round.
- type 2: a round with the two calls $(0, 1)$ and $(3, 2)$. Note that there is at most one such round since only one message (namely that of 0) is transmitted on the arc $(0, 1)$.
- type 3: a round with the unique 2-call $\{2 : 1, 3\}$.
- type 4: a round with the two calls $(1, 0)$ and $(2, 3)$.

For the rounds of type 3 and 4, we note that only three messages (namely messages $0, 1, 2$) are transmitted via the call $(2, 3)$ and so we have at most 3 rounds of the last two types (one of type 3 and two of type 4, or three of type 4). So altogether among the rounds needed to transmit the $3n$ messages using one of the 6 calls $(0, 1)$, $(1, 0)$, $(1, 2)$, $(2, 1)$, $(2, 3)$, and $(3, 2)$, we have at most 5 rounds with two of these 6 calls. So for $n \geq 4$, we need at least $3n - 5$ rounds. ⋄

3. Upper Bound for $n \geq 4$

Let $p = \lceil n/2 \rceil$, $n \geq 4$.

We will design in this section a protocol with $3n - 5$ rounds. In fact we will give an exhaustive and explicit description of each round r, $1 \leq r \leq 3n-5$, by giving the set of calls contained in the round. We will have 3 different phases in the protocol, where the rounds will be grouped into sequences. In phase 1, we have a sequence S_0 of 3 rounds in which we will use all the 2-calls. Then, in phase 2, we have $p - 1$ sequences of 4 rounds constructed in a regular algorithmic manner and we will prove that at the end of phase 2, all messages will arrive in the middle node p, while nodes in the left part i.e., nodes $i < p$ (resp. nodes in the right part $i > p$) have received all the messages $j < i$ (resp. $j > i$). Phase 3 will contain the last $3n - 5 - 4p + 1$ rounds grouped in sequences of 2 rounds. The protocol and the proof of its validity will depend on the parity of n.

Note that though the protocol has some regularities, it is involved and the proof of its validity is not easy. So it might be helpful for this reason to first see that the protocols work well on some examples. We will explain how to read the examples given in the tables at the end of the article and point out some features of the protocol and some of the difficulties we have encountered. (Precise description of the rounds will be given in detail afterwards).

Let us first follow the protocol on the examples given for $n = 12, 13, 14, 15$ (in **Table A·1**, **Table A·2**, **Table A·3**, and **Table A·4**). In these tables, line r corresponds to the round r, and column i to the node i. To facilitate the reading of the tables, we indicate in the cell (r, i) the value of the message received in round r by the node i. If the cell (r, i) is empty, it means that node i is sending a message in round r (and so cannot receive one). If the cell (r, i) contains a cross, it means that node i is neither sending nor receiving at round r. In the description of the protocol and in the proof we will give the set of calls contained in round r. In the figures, the set of calls of a given round is not explicitly indicated, but it can be easily deduced. Indeed if in the cell (r, i) there is a value $j > i$ (respectively $j < i$) then it implies that round r contains the call $(i + 1, i)$ (resp. the call $(i - 1, i)$). Furthermore, we have indicated in the second column the value of the sequence considered and in the third column the set of calls of the round r (when it is written as a set B_k in a round of the sequence s, it corresponds to the set $B_k(s)$ described in the text).

As an example, consider the case $n = 12$ and the round $r = 4$. In the cell $(4, 2)$, there is a value 4 and so the round 4 contains the call $(3, 2)$ (transmitting the message 4 to node 2). In the cell $(4, 5)$, there is a 3, and this implies that round 4 contains the call $(4, 5)$ (transmitting the message 3 to node 5). Similarly round 4 contains the calls $(7, 6)$ and $(8, 9)$. Nodes $3, 4, 7, 8$ are sending and so the corresponding cells are empty. Nodes $0, 1, 10, 11$ are neither sending nor receiving so there is a cross in each of the corresponding cells. In summary the round 4 consists of the calls $(3, 2)$, $(4, 5)$, $(7, 6)$ and $(8, 9)$.

In the 3 first rounds (phase 1 or sequence S_0) all the 2-calls appear. For example, in round 1 there is a 1 in cell $(1, 0)$ and so round 1 contains the call $(1, 0)$, while in cell $(0, 2)$ there is a 1 and this implies that round 1 also contains the call $(1, 2)$. These 2 calls form the 2-call $\{1 : 0, 2\}$.

We note that these rounds do not necessarily have the maximum number of possible calls like round 4 for $n = 12$. In fact the set of calls of this round is a subset of the set $A_0 = \{(0, 1), (3, 2), (4, 5), (7, 6), (8, 9), (11, 10)\}$, but calls $(0, 1)$ and $(11, 10)$ are useless since they have no message to transmit. However, we note that there is a regularity for the nodes in the middle such as 5 and 6. Indeed, in phase 2, in each sequence of 4 rounds S_s (containing rounds $4s, 4s + 1, 4s + 2, 4s + 3$) with $s = 1, 2, 3, 4$, node 5 (or 6) receives one message from its left neighbor, one message from its right neighbor and sends both to its left and right neighbors. That is the same for the sequence S_5 except in round 21 there is no call $(4, 5)$, as node 5 already receives all the messages $j \leq 4$. So at the end of phase 2 (end of round 23) nodes 5 and 6 have received all the messages. The situation is different for nodes in the left part (or right part). For example consider node 0. It has only its own message to send and it was sent in round 3. So in further rounds there will be no more useful calls via $(0, 1)$. In contrary, node 0 should receive messages from 1 to $n - 1$ via the call $(1, 0)$. In order to have the protocol end with a total number of $3n - 5$ rounds, node 0 should receive on average one message every 3 rounds. That is correct for the sequence S_0 of phase 1 consisting of 3 rounds, however the other sequences in phase 2 have 4 rounds. To obtain this average of 3 receptions, we impose 4 receptions during 3 consecutive sequences. For example, node 0 should receive only one message in 2 sequences, and two messages in the third sequence (in the protocol, node 0 receives two messages in sequence S_2 and S_5 and so on). We can see the difference between n even and odd. If n is even, node 0 receives two messages in rounds 9 and 10 of the sequence S_2, while if n is odd, it receives two messages in rounds 8 and 11.

Other difficulties and tools will be described in the following text. Already we note the importance of the order of the rounds inside a sequence of phase 2. For example in the sequence S_2, the round containing the set of calls $B_3(2)$ (indicated in the example as round 8) should be completed before the round containing the set of calls $B_2(2)$ (indicated as round 10). Otherwise, if we do in the sequence S_2 the rounds $B_1(2)$ and $B_2(2)$ before the round $B_3(2)$, we cannot transmit the message 4 via the call $(1, 0)$ since it will not arrive at node 1 on time. We can check in the figures that when a message j is sent in some round r to node i via the call $(i + 1, i)$, it has already arrived in node $i + 1$ in a round before r. So we have to describe not only the calls included in a round but also verify that they have some new message to transmit and therefore a very careful analysis is needed.

Remark that in what follows we will assume that the nodes mentioned are all in the range $[0, n - 1]$. For example, if a node $i - 1$ is used, then implicitly we assume that $1 \leq i \leq n - 1$.

3.1 Phase 1 (Sequence S_0)

In the first 3 rounds we do all the 2-calls $\{i : i - 1, i + 1\}$. For the two end nodes 0 and $n - 1$, the 2-calls are reduced to the regular calls $(0, 1)$ and $(n - 1, n - 2)$. More precisely, the rounds $r = 1, 2, 3$ of the sequence S_0 consist of the sets R_r of 2-calls $R_r = \{3j + r : 3j + r - 1, 3j + r + 1\}$ for $0 \leq 3j + r \leq n - 1$.

Claim 2 *After the first 3 rounds each node $1 \leq i \leq n - 2$ has received the two messages $i - 1$ and $i + 1$. Node 0 has received the message 1 and node $n - 1$ the message $n - 2$.*

Proof. The claim follows from the fact that in one of the 3 rounds, each node i, $0 \leq i \leq n - 1$ is the sender of a 2-call, while in another round node i, $1 \leq i \leq n - 1$ is the receiver of the 2-call with sender $i - 1$ and in another round node i, $0 \leq i \leq n - 2$ is the receiver of the 2-call with sender $i + 1$. ◇

3.2 Phase 2

Here, we will do sequences S_s of 4 rounds $4s, 4s + 1, 4s + 2$, and $4s + 3$ for $1 \leq s \leq p - 1$. The first idea is to make the maximum number of alternating calls. More precisely, if $(i, i+1)$ (resp. $(i, i - 1)$) is a call in a round, then so is the call $(i + 3, i + 2)$ (resp. $(i + 1, i + 2)$), if it exists.

Recall that a round is defined by the set of (compatible) calls it contains. In what follows we use the shortened notation “round A” to mean “round containing the set A of calls”. We will mainly use rounds obtained from the following 4 rounds A_k, $0 \leq k \leq 3$, where the subscript k indicates that node k sends to the right via the call $(k, k + 1)$.

- Round A_0 contains the set A_0 of calls $(4j, 4j + 1)$, $0 \leq 4j \leq n - 2$ and $(4j + 3, 4j + 2)$, $0 \leq 4j \leq n - 4$.
- Round A_1 contains the set A_1 of calls $(4j + 1, 4j + 2)$, $0 \leq 4j \leq n - 3$ and $(4j + 4, 4j + 3)$, $0 \leq 4j \leq n - 5$.
- Round A_2 contains the set A_2 of calls $(4j + 2, 4j + 3)$, $0 \leq 4j \leq n - 4$ and $(4j + 1, 4j)$, $0 \leq 4j \leq n - 2$.
- Round A_3 contains the set A_3 of calls $(4j + 3, 4j + 4)$, $0 \leq 4j \leq n - 5$ and $(4j + 2, 4j + 1)$, $0 \leq 4j \leq n - 3$.

Note that rounds A_0 and A_2 (and also A_1 and A_3) contain symmetric calls. Furthermore, during the 4 rounds, each node different from $0, 1, n-2, n-1$, is exactly once a receiver from the left, once a receiver from the right, once a sender to the left, and once a sender to the right.

3.2.1 Sequence S_1

The sequence S_1 (rounds 4 to 7) will consist of the set of 4 rounds $\{A_0, A_1, A_2, A_3\}$. We can do them in any order not necessarily (A_0, A_1, A_2, A_3).

Claim 3 *During the sequence S_1 (rounds 4 to 7), each node $2 \le i \le n-3$ receives messages $i-2$ and $i+2$. Node 0 receives message 2, node 1 receives message 3, node $n-2$ receives message $n-4$, and node $n-1$ message $n-3$.*

Proof. The claim follows from the fact that each node i (except $0, 1, n-2, n-1$) receives in one of the 4 rounds message $i-2$ and in another round, message $i+2$. For example node 3 receives message 1 via the call $(2, 3)$ in round A_2 and message 5 via the call $(4, 3)$ in round A_1. Note that during sequence S_1, the calls $(0, 1)$ and $(n-1, n-2)$ are useless since node 0 (resp. $n-1$) has no message to send to 1 (resp. $n-2$). Therefore, node 1 receives only message 3 and node $n-2$ only message $n-4$. Finally, node 0 (resp. $n-1$) receives only message 2 (resp. $n-3$) because it is an end node of the chain. ⋄

3.2.2 Idea of the Protocol for a General Sequence S_s

A simple protocol will consist in repeating the sequence S_1, a total of $n-3$ times. This protocol will complete in $4n-5$ rounds, which is not optimal. This is not surprising since many calls become useless in the process, specifically in the sth sequence these are the calls $(i-1, i)$ with $i \le s$ and $(j+1, j)$ with $j \ge n-s$. For example, for $s = 2$, calls $(0, 1), (1, 2), (n-2, n-3), (n-1, n-2)$ are useless. Therefore, we will construct the sequence S_s by keeping only the middle part of the set A_k and deleting useless calls and adding some valid calls on both sides. These added calls will be all directed to the left (resp. right) in the left (resp. right) part. However we will see that in order that these modifications give valid rounds, we will have to choose a suitable order for the modified rounds.

Let us first define the sequence S_2 (rounds 8 to 11) and then S_3 (rounds 12 to 15) before defining the general sequence S_s.

3.2.3 Sequence S_2

Note that call $(1, 2)$ is now useless since node 1 has no new message to transmit to node 2. We know that call $(1, 2)$ appears in round A_1. So we will delete the call $(1, 2)$ in A_1 and add a call $(1, 0)$ and this will bring a message to node 0. We do the same modification for the useless call $(n-2, n-3)$. More precisely, if we let $\boldsymbol{n \equiv \gamma}$ **(mod 4)**, then the call $(n-2, n-3)$ appears in the round $A_{\gamma-1}$.

For example, for $n = 12$, call $(10, 9)$ appears in round A_3, while for $n = 13$, call $(11, 10)$ appears in round A_0. We will delete the call $(n-2, n-3)$ in $A_{\gamma-1}$ and add a call $(n-2, n-1)$ which will bring a message to node $n-1$.

In the rest of the paper, we will use the notation $B_k(s)$ for the 4 rounds of the sequence S_s where the value k is always taken modulo 4. We first construct $B_k(2)$ by modifying the A_k as explained above. Then we will see that the order in which we do the 4 rounds of S_2 is important and that only some orders are valid.

Construction of the rounds $B_k(2)$ of S_2

$B_0(2)$ is obtained from A_0 by deleting call $(0, 1)$. Furthermore, when $\gamma = 1$, we also delete call $(n-2, n-3)$ and add call $(n-2, n-1)$, and when $\gamma = 0$, we delete $(n-1, n-2)$.

$B_1(2)$ is obtained from A_1 by deleting call $(1, 2)$ and adding call $(1, 0)$. Furthermore, when $\gamma = 2$, we also delete call $(n-2, n-3)$ and add call $(n-2, n-1)$, and when $\gamma = 1$, we delete $(n-1, n-2)$.

$B_2(2) = A_2$, except when $\gamma = 3$, we delete call $(n-2, n-3)$ and add call $(n-2, n-1)$, and when $\gamma = 2$, we delete $(n-1, n-2)$.

$B_3(2) = A_3$, except when $\gamma = 0$, we delete call $(n-2, n-3)$ and add call $(n-2, n-1)$, and when $\gamma = 3$, we delete $(n-1, n-2)$.

Constraints on the order of the rounds $B_k(2)$ of the sequence S_2

In the sequence S_2, we now have two calls $(1, 0)$, one in round $B_2(2)$ and the other that was added in round $B_1(2)$, and these should transmit two new messages to node 0 namely messages 3 and 4. Node 1 knows the message 3 at the end of sequence S_1, but it receives message 4 only in round $B_3(2)$. So the order in which we will do the 4 rounds of sequence S_2 is important. For example, the order $(B_0(2), B_1(2), B_2(2), B_3(2))$ will not be valid. In a valid order, round $B_3(2)$ should be completed before at least one of the two rounds $B_1(2)$ and $B_2(2)$. We express this fact by noting that the order $\prec$ on the rounds should satisfy the following constraint.

$$B_3(2) \prec max\{B_1(2), B_2(2)\}$$

Similarly, in the sequence S_2, we now have two calls $(n-2, n-1)$ in rounds $B_{\gamma-1}(2)$ and $B_{\gamma-2}(2)$ (where we recall that $n \equiv \gamma$ (mod 4) and the subscripts of the B are integers modulo 4). These two calls should transmit two new messages to node $n-1$, namely messages $n-4$ and $n-5$. Node $n-2$ knows the message $n-4$ at the end of sequence S_1, but it receives message $n-5$ only in round $B_{\gamma-3}(2)$ and so the order $\prec$ on the rounds should satisfy the following constraint.

$$B_{\gamma-3}(2) \prec max\{B_{\gamma-1}(2), B_{\gamma-2}(2)\}$$

Note that if the two constraints above are satisfied, all the calls in any round are valid. There are many orders satisfying these two constraints (see analysis for the general case). We can choose the following orders (used in the tables for $n = 12, 13, 14, 15$):

$$\begin{cases} (B_3(2), B_1(2), B_2(2), B_0(2)) \text{ if } n \text{ is even } (\gamma = 0 \text{ or } 2) \\ (B_2(2), B_3(2), B_0(2), B_1(2)) \text{ if } n \text{ is odd } (\gamma = 1 \text{ or } 3) \end{cases}$$

Messages received during the sequence S_2

We summarize the status of messages received in sequence S_2 in the following claim.

Claim 4 *There exists an order of the 4 rounds $B_k(2)$ of sequence S_2 (rounds 8 to 11) such that, during the sequence S_2, each node $3 \le i \le n-2$ has received message $i-3$, each node $1 \le i \le n-4$ has received message $i+3$, node 0 has received messages 3 and 4, and node $n-1$ has received messages $n-4$ and $n-5$.*

Proof. The claim follows from the fact that for node $3 \le i \le n-4$, the calls are those of the A_k and so each node in one of the 4 rounds receives a new message from the left, namely message

$i-3$ and in another round receives a new message from the right, namely message $i+3$. Node 1 (resp. 2) receives only message 4 (resp. 5) from the right and node $n-3$ (resp. $n-2$) receives only message $n-6$ (resp. $n-5$) from the left. As we have seen above, node 0 (resp. $n-1$) receives two messages 4 and 5 (resp. $n-4$ and $n-5$). But that is possible only if the order $\prec$ on the rounds satisfies the two constraints given above. ⋄

3.2.4 Sequence S_3

Construction of the rounds $B_k(3)$ of S_3

We have now more useless calls such as $(0,1),(1,2),(2,3),(n-1,n-2),(n-2,n-3),(n-3,n-4)$. We will delete them, and add some more calls as in S_2. In $B_k(3)$ we will keep only the calls of A_k involving nodes inside $[4,n-5]$. We denote these calls in the middle part by $A_k[4,n-5]$.

For example, for $s=3$ and $n=13$:

$$A_0[4,8]=\{(4,5),(7,6),(8,9)\},$$
$$A_1[4,8]=\{(4,3),(5,6),(8,7)\},$$
$$A_2[4,8]=\{(5,4),(6,7),(9,8)\},$$
$$A_3[4,8]=\{(3,4),(6,5),(7,8)\}.$$

The rounds $B_k(3)$ will be in the form

$$B_k(3)=\{L_k(3),A_k[4,n-5],R_k(n-4)\}$$

where the indices k are taken modulo 4.

The left part $L_k(3)$ is defined to be a set containing the call $(i,i-1)$ $(i\le 3)$, which does not interfere with the call of $A_k[4,n-5]$ involving node 4, and where, furthermore, i is chosen to be the maximum. Similarly the right part $R_k(n-4)$ is defined to be a set containing the call $(j,j+1)$ $(j\ge n-4)$ which does not interfere with the call of $A_k[4,n-5]$ involving node $n-5$, and where, furthermore, j is chosen to be the minimum. Therefore, as call $(4,5)$ appears in $A_0[4,n-5]$, $L_0(3)=(3,2)$, as call $(4,3)$ appears in $A_1[4,n-5]$, $L_1(3)=(1,0)$, as call $(5,4)$ appears in $A_2[4,n-5]$, $L_2(3)=(2,1)$, and as call $(3,4)$ appears in $A_3[4,n-5]$, $L_3(3)=(2,1)$.

Similarly (recall that $n\equiv\gamma\ (\mathrm{mod}\ 4)$), as call $(n-5,n-6)$ appears in $A_\gamma[4,n-5]$, $R_\gamma(n-4)=(n-4,n-3)$, as call $(n-4,n-5)$ appears in $A_{\gamma+1}[4,n-5]$, $R_{\gamma+1}(n-4)=(n-3,n-2)$, as call $(n-6,n-5)$ appears in $A_{\gamma+2}[4,n-5]$, $R_{\gamma+2}(n-4)=(n-3,n-2)$, and as call $(n-5,n-4)$ appears in $A_{\gamma-1}[4,n-5]$, $R_{\gamma-1}(n-4)=(n-2,n-1)$.

For example for $s=3$ and $n=13$ $(\gamma=1)$ we get:

$$B_0(3)=\{(3,2),(4,5),(7,6),(8,9),(11,12)\},$$
$$B_1(3)=\{(1,0),(4,3),(5,6),(8,7),(9,10)\},$$
$$B_2(3)=\{(2,1),(5,4),(6,7),(9,8),(10,11)\},\text{ and}$$
$$B_3(3)=\{(2,1),(3,4),(6,5),(7,8),(10,11)\}.$$

Constraints on the order of the rounds $B_k(3)$ of the sequence S_3

In the sequence S_3, we now have two calls $(2,1)$ in rounds $B_2(3)$ and $B_3(3)$, which should transmit two messages to node 1 namely messages 5 and 6. Node 2 knows the message 5 at the end of sequence S_2, but it receives message 6 only in round $B_0(3)$. So $B_0(3)$ should be completed before at least one of the two rounds $B_2(3)$ and $B_3(3)$. Therefore the following constraint should be satisfied.

$$B_0(3)\prec max\{B_2(3),B_3(3)\}$$

Node 1 does not know at the end of S_2 the message 5, but it should transmit it to node 0 in round $B_1(3)$. It receives this message in the first of the two rounds $\{B_2(3),B_3(3)\}$. So at least one of the two rounds $B_2(3)$ and $B_3(3)$ should be completed before $B_1(3)$. We express this fact by noting that the order $\prec$ on the rounds should satisfy the following constraint.

$$min\{B_2(3),B_3(3)\}\prec B_1(3)$$

Similarly, in the sequence S_3, we now have

- two calls $(n-3,n-2)$ in rounds $B_{\gamma+1}(3)$ and $B_{\gamma+2}(3)$ which should transmit two messages to node $n-2$ namely messages $n-6$ and $n-7$,
- one call $(n-4,n-3)$ in round $B_\gamma(3)$ which should transmit message $n-7$, and
- one call $(n-2,n-1)$ in round $B_{\gamma-1}(3)$ which should transmit message $n-6$.

Node $n-3$ knows the message $n-6$ at the end of sequence S_2, but it receives message $n-7$ only in round $B_\gamma(3)$. So $B_\gamma(3)$ should be completed before at least one of the two rounds $B_{\gamma+1}(3)$ and $B_{\gamma+2}(3)$. Therefore we should have the following.

$$B_\gamma(3)\prec max\{B_{\gamma+1}(3),B_{\gamma+2}(3)\}$$

Node $n-2$ does not know at the end of S_2 the message $n-6$, but should transmit it in round $B_{\gamma-1}(3)$. It receives this message in the first of the two rounds $\{B_{\gamma+1}(3),B_{\gamma+2}(3)\}$. So at least one of the two rounds $B_{\gamma+1}(3)$ and $B_{\gamma+2}(3)$ should be completed before $B_{\gamma-1}(3)$. Therefore we should have the following.

$$min\{B_{\gamma+1}(3),B_{\gamma+2}(3)\}\prec B_{\gamma-1}(3)$$

Note that there are many orders satisfying the four constraints above (see analysis for the general case). We can choose for example the following orders according to the values of n.

$$\begin{cases}(B_0(3),B_2(3),B_3(3),B_1(3)) \text{ if } n \text{ is even } (\gamma=0 \text{ or } 2)\\ (B_3(3),B_0(3),B_1(3),B_2(3)) \text{ if } n \text{ is odd } (\gamma=1 \text{ or } 3)\end{cases}$$

Messages received during the sequence S_3

We summarize the status of messages received in sequence S_3 in the following claim.

Claim 5 *There exists an order of the 4 rounds $B_k(3)$ of sequence S_3 (rounds 12 to 15) (for example those defined above), such that during the sequence S_3, each node $4\le i\le n-3$ has received message $i-4$, each node $2\le j\le n-5$ message $j+4$, node 0 message 5, node 1 messages 5 and 6, node $n-1$ message $n-6$, and node $n-2$ messages $n-6$ and $n-7$.*

Proof. The claim follows from the fact that for node $4\le i\le n-5$, the calls involved are those of A_k, and so node i receives a new message from the left, namely message $i-4$ in one of the 4 rounds and receives a new message from the right, namely message $i+4$ in another round. Node 2 (resp. 3) receives only message 6 (resp. 7) from the right and node $n-4$ (resp. $n-3$) receives only message

$n-8$ (resp. $n-7$) from the left. As we have seen above, node 1 (resp. $n-2$) receives two messages 5 and 6 (resp. $n-6$ and $n-7$). But that is possible only if the order $\prec$ on the rounds satisfies the two "max-constraints" given above. Node 0 (resp. $n-1$) receives message 5 (resp. $n-6$) but this is possible only if the order $\prec$ on the rounds satisfies the two "min-constraints" given above. In summary, for any order satisfying the four constraints (see example above), the claim is true. $\diamond$

3.2.5 Sequence S_s

Just as for $s = 3$, the rounds $B_k(s)$ will consist of 3 parts: one left part $L_k(s)$with a set of calls all directed to the left, a middle part $A_k[s+1, n-2-s]$, and a right part $R_k(n-1-s)$ with a set of calls all directed to the right. We will have similar constraints on the orders of the rounds and we will see that there exist two canonical orders according to the parity of n. We next precisely define these 3 parts.

Construction of the rounds $B_k(s)$ of S_s

For a general s, we note (see Claim 6) that at the end of the sequence S_{s-1}, the nodes $1 \le i \le s$ have received all the messages from the left (that is messages $j \le i$), while nodes $n-s-1 \le i \le n-2$ have received all the messages from the right (that is messages $j \ge i$). Therefore, in the rounds A_k, such as for $s = 2, 3$, there are many useless calls in particular the calls $(s-1, s)$ and $(n+1-s, n-s)$ which were useful in the preceding sequence. So in $B_k(s)$, we will keep only the set of calls of A_k with a sender or a receiver in the interval $[s+1, n-2-s]$, denoted by $A_k[s+1, n-2-s]$.

For example for $s = 11$ and $n = 32$,

$$A_0[12, 19] = \{(12, 13), (15, 14), (16, 17), (19, 18)\},$$
$$A_1[12, 19] = \{(12, 11), (13, 14), (16, 15), (17, 18), (20, 19)\},$$
$$A_2[12, 19] = \{(13, 12), (14, 15), (17, 16), (18, 19)\},$$
$$A_3[12, 19] = \{(11, 12)(14, 13), (15, 16), (18, 17), (19, 20)\}.$$

We will do the sequence S_s till $s = p-1$. For $s = p-1$, when $n = 2p+1$ is odd, then $s+1 = n-2-s$ and the interval $[s+1, n-2-s]$ is reduced to the node p. For $s = p-1$, when $n = 2p$ is even, then $s+1 > n-2-s$ and in this particular case the middle part will be empty.

Having defined the set of calls in the middle part, we now construct the calls in the left (resp. right) part of $B_k(s)$ denoted by $L_k(s)$ (resp. $R_k(n-1-s)$). Here, $B_k(s)$ is obtained by the concatenation of these three sets

$$B_k(s) = \{L_k(s), A_k[s+1, n-2-s], R_k(n-1-s)\}$$

Recall that all the indices k are taken modulo 4.

For the left part, in order to have the maximum number of calls, we will first put in $L_k(s)$ the call $(i_{max}, i_{max}-1)$, where i_{max} is the greatest integer $\le s$ such that the call $(i_{max}, i_{max}-1)$ does not interfere with the call in $A_k[s+1, n-2-s]$ involving node $s+1$. Then we add in $L_k(s)$ the calls $(i_{max}-3j, i_{max}-3j-1)$ for $0 \le 3j \le i_{max}-1$. These calls are not pairwise interfering since nodes $i_{max}-3j-2$ do nothing (such idle nodes are indicated by an $\times$ in the tables). In the example given before with $s = 11$, the call of $A_0[12, 19]$ involving node 12 is $(12, 13)$, so $i_{max} = 11$ and we get $L_0(11) = \{(11, 10), (8, 7), (5, 4), (2, 1)\}$.

Let $s \equiv \alpha$ (mod 4).

The call $(s+2, s+1)$ appears in $A_{\alpha-1}[s+1, n-2-s]$. In that case, $i_{max} = s-1$ and we get

$$L_{\alpha-1}(s) = \{(s-3j-1, s-3j-2)\} \quad 0 \le 3j \le s-2.$$

The call $(s, s+1)$ appears in $A_\alpha[s+1, n-2-s]$. In that case, we also have $i_{max} = s-1$ and so $L_\alpha(s) = L_{\alpha-1}(s)$ and we get

$$L_\alpha(s) = \{(s-3j-1, s-3j-2)\} \quad 0 \le 3j \le s-2.$$

The call $(s+1, s+2)$ appears in $A_{\alpha+1}[s+1, n-2-s]$. In that case, $i_{max} = s$ and we get

$$L_{\alpha+1}(s) = \{(s-3j, s-3j-1)\} \quad 0 \le 3j \le s-1.$$

The call $(s+1, s)$ appears in $A_{\alpha+2}[s+1, n-2-s]$. In that case, $i_{max} = s-2$ and we get

$$L_{\alpha+2}(s) = \{(s-3j-2, s-3j-3)\} \quad 0 \le 3j \le s-3.$$

In the example with $s = 11$ ($\equiv 3 \pmod 4$, or $\alpha = 3$), we have:

$$L_2(11) = \{(10, 9), (7, 6), (4, 3), (1, 0)\},$$
$$L_3(11) = \{(10, 9), (7, 6), (4, 3), (1, 0)\},$$
$$L_0(11) = \{(11, 10), (8, 7), (5, 4), (2, 1)\}, \text{ and}$$
$$L_1(11) = \{(9, 8), (6, 5), (3, 2)\}.$$

For the right part, we do a similar construction obtained by symmetry (node i is replaced by the node $n-1-i$ and the calls are in the opposite direction). More precisely, in order to have the maximum number of calls, we will first put the call $(i_{min}, i_{min}+1)$ in $R_k(n-1-s)$, where i_{min} is the smallest integer $\ge n-1-s$ such that the call $(i_{min}, i_{min}+1)$ does not interfere with the call in $A_k[s+1, n-2-s]$ involving node $n-2-s$. Then we add in $R_k(n-1-s)$ the calls $(i_{min}+3j, i_{min}+3j+1)$ for $0 \le 3j \le n-2-i_{min}$. These calls are not pairwise interfering since the nodes $i_{min}+3j+2$ do nothing (such idle nodes are indicated by an $\times$ in the tables). In the example given before with $s = 11$, $n = 32$ and so $n-2-s = 19$, the call of $A_0[12, 19]$ involving node 19 is $(19, 18)$. Therefore $i_{min} = 20$ and we get $R_0(20) = \{(20, 21), (23, 24), (26, 27), (29, 30)\}$.

Let $n-1-s \equiv \beta$ (mod 4). (In the preceding subsections we use $n \equiv \gamma \pmod 4$, so for $s = 2$, $\gamma = \beta - 1$ and for $s = 3$, $\gamma = \beta$).

The call $(n-s-3, n-s-2)$ appears in $A_{\beta+2}[s+1, n-2-s]$. In that case, $i_{min} = n-s$ and we get

$$R_{\beta+2}(n-1-s) = \{(n-s+3j, n-s+3j+1)\} \quad 0 \le 3j \le s-2.$$

The call $(n-s-1, n-s-2)$ appears in $A_{\beta+1}[s+1, n-2-s]$. Here again $i_{min} = n-s$ and so, $R_{\beta+1}(n-1-s) = R_{\beta+2}(n-1-s)$ and we get

$$R_{\beta+1}(n-1-s) = \{(n-s+3j, n-s+3j+1)\} \quad 0 \le 3j \le s-2.$$

The call $(n-s-2, n-s-3)$ appears in $A_\beta[s+1, n-2-s]$. In that case, $i_{min} = n-s-1$ and we get

$$R_\beta(n-1-s) = \{(n-s+3j-1, n-s+3j)\} \quad 0 \le 3j \le s-1.$$

Journal of Information Processing Vol.28 889–902 (Dec. 2020)

The call $(n-s-2, n-s-1)$ appears in $A_{\beta-1}[s+1, n-2-s]$. In that case, $i_{min} = n-s+1$ and we get

$$R_{\beta-1}(n-1-s) = \{(n-s+3j+1, n-s+3j+2)\} \quad 0 \leq 3j \leq s-3.$$

In the example with $n = 32$ and $s = 11$, $n-1-s = 20$ ($\equiv 0 \pmod 4$, or $\beta = 0$) we have:

$$R_2(20) = \{(21,22),(24,25),(27,28),(30,31)\}$$
$$R_1(20) = \{(21,22),(24,25),(27,28),(30,31)\},$$
$$R_0(20) = \{(20,21),(23,24),(26,27),(29,30)\}, \text{ and}$$
$$R_3(20) = \{(22,23),(25,26),(28,29)\}.$$

If we concatenate the values obtained for the example for $L_k(11), A_k[12,19]$, and $R_k(20)$ we get the following $B_k(s)$. $B_0(11) = \{L_0(11), A_0[12,19], R_0(20)\} = \{(11,10),(8,7),(5,4),(2,1),(12,13),(15,14),(16,17),(19,18),(20,21),(23,24),(26,27),(29,30)\}$.

$B_1(11) = \{L_1(11), A_1[12,19], R_1(20)\} = \{(9,8),(6,5),(3,2),(12,11),(13,14),(16,15),(17,18),(20,19),(21,22),(24,25),(27,28),(30,31)\}$.

$B_2(11) = \{L_2(11), A_2[12,19], R_2(20)\} = \{(10,9),(7,6),(4,3),(1,0),(13,12),(14,15),(17,16),(18,19),(21,22),(24,25),(27,28),(30,31)\}$.

$B_3(11) = \{L_3(11), A_3[12,19], R_3(20)\} = \{(10,9),(7,6),(4,3),(1,0),(11,12),(14,13),(15,16),(18,17),(19,20),(22,23),(25,26),(28,29)\}$.

Constraints on the order of the rounds $B_k(s)$ of the sequence S_s

Just as for the case $s = 3$, for $s \equiv \alpha \pmod 4$, the calls $\{(s-3j-1, s-3j-2)\}$, $0 \leq 3j \leq s-2$ appear twice namely in rounds $B_{\alpha-1}(s)$ and $B_\alpha(s)$ in which two messages should be transmitted. But node $s-3j-1$ has only one message and receives the second one via the call $\{(s-3j, s-3j-1)\}$ in round $B_{\alpha+1}(s)$, and so we have the following constraint on the orders.

$$B_{\alpha+1}(s) \prec max\{B_{\alpha-1}(s), B_\alpha(s)\}$$

Furthermore, in round $B_{\alpha+2}(s)$, node $s-3j-2$ has to transmit the message received via one of the two calls $\{(s-3j-1, s-3j-2)\}$ and so we have the second constraint.

$$min\{B_{\alpha-1}(s), B_\alpha(s)\} \prec B_{\alpha+2}(s)$$

Similarly, in the right part, for $n-1-s \equiv \beta \pmod 4$, the calls $\{(n-s+3j, n-s+3j+1)\}$ $(0 \leq 3j \leq s-2)$ appear twice, namely in rounds $B_{\beta+2}(s)$ and $B_{\beta+1}(s)$ in which two messages should be transmitted. But node $n-s-3j$ has only one message and it receives the second one via the call $\{(n-s-3j-1, n-s-3j)\}$ in round $B_\beta(s)$. So we have the following constraint on the orders.

$$B_\beta(s) \prec max\{B_{\beta+1}(s), B_{\beta+2}(s)\}$$

Furthermore, in round $B_{\beta-1}(s)$, nodes $n-s+3j+1$ $(0 \leq 3j \leq s-2)$ have to transmit the message received via one of the two calls $\{(n-s+3j, n-s+3j+1)\}$, and so we have the second constraint.

$$min\{B_{\beta+1}(s), B_{\beta+2}(s)\} \prec B_{\beta-1}(s)$$

Let us now determine the orders that satisfy the 4 constraints above.

Recall that $s \equiv \alpha \pmod 4$ and $n-1-s \equiv \beta \pmod 4$. We will see that there are two cases: β has the same parity as α which happens when n is odd, and β has a different parity as α which happens when n is even.

- When n is even, then β has a different parity from α.
  If $\beta \equiv \alpha + 1 \pmod 4$ or $\beta \equiv \alpha + 3 \pmod 4$, we have four orders which satisfy the 4 constraints as follows:

  $$(B_{\alpha+1}(s), B_{\alpha-1}(s), B_\alpha(s), B_{\alpha+2}(s)),$$
  $$(B_{\alpha+1}(s), B_{\alpha-1}(s), B_{\alpha+2}(s), B_\alpha(s)),$$
  $$(B_{\alpha-1}(s), B_{\alpha+1}(s), B_\alpha(s), B_{\alpha+2}(s)),$$
  $$(B_{\alpha-1}(s), B_{\alpha+1}(s), B_{\alpha+2}(s), B_\alpha(s)).$$

  We choose the first one for n even, then show it with the value of s (and α).

  $$(B_1(s), B_3(s), B_0(s), B_2(s)) \text{ for } s \equiv 0 \pmod 4\ (\alpha = 0)$$
  $$(B_2(s), B_0(s), B_1(s), B_3(s)) \text{ for } s \equiv 1 \pmod 4\ (\alpha = 1)$$
  $$(B_3(s), B_1(s), B_2(s), B_0(s)) \text{ for } s \equiv 2 \pmod 4\ (\alpha = 2)$$
  $$(B_0(s), B_2(s), B_3(s), B_1(s)) \text{ for } s \equiv 3 \pmod 4\ (\alpha = 3)$$

- When n is odd, then β has the same parity as α.
  If $\beta \equiv \alpha \pmod 4$, we have six orders which satisfy the 4 constraints as follows:

  $$(B_\alpha(s), B_{\alpha+1}(s), B_{\alpha+2}(s), B_{\alpha-1}(s)),$$
  $$(B_\alpha(s), B_{\alpha+1}(s), B_{\alpha-1}(s), B_{\alpha+2}(s)),$$
  $$(B_{\alpha+1}(s), B_\alpha(s), B_{\alpha+2}(s), B_{\alpha-1}(s)),$$
  $$(B_{\alpha+1}(s), B_\alpha(s), B_{\alpha-1}(s), B_{\alpha+2}(s)),$$
  $$(B_{\alpha+1}(s), B_{\alpha-1}(s), B_\alpha(s), B_{\alpha+2}(s)),$$
  $$(B_\alpha(s), B_{\alpha+2}(s), B_{\alpha+1}(s), B_{\alpha-1}(s)).$$

  If $\beta \equiv \alpha + 2 \pmod 4$, we have four orders which satisfy the 4 constraints as follows:

  $$(B_\alpha(s), B_{\alpha+1}(s), B_{\alpha+2}(s), B_{\alpha-1}(s)),$$
  $$(B_\alpha(s), B_{\alpha+2}(s), B_{\alpha+1}(s), B_{\alpha-1}(s)),$$
  $$(B_{\alpha-1}(s), B_{\alpha+1}(s), B_{\alpha+2}(s), B_\alpha(s)),$$
  $$(B_{\alpha-1}(s), B_{\alpha+2}(s), B_{\alpha+1}(s), B_\alpha(s)).$$

  We select one of these orders that applies to both cases (the first one), and show it with the value of s (and α).

  $$(B_0(s), B_1(s), B_2(s), B_3(s)) \text{ for } s \equiv 0 \pmod 4\ (\alpha = 0)$$
  $$(B_1(s), B_2(s), B_3(s), B_0(s)) \text{ for } s \equiv 1 \pmod 4\ (\alpha = 1)$$
  $$(B_2(s), B_3(s), B_0(s), B_1(s)) \text{ for } s \equiv 2 \pmod 4\ (\alpha = 2)$$
  $$(B_3(s), B_0(s), B_1(s), B_2(s)) \text{ for } s \equiv 3 \pmod 4\ (\alpha = 3)$$

Messages received during the sequence S_s

We summarize the status of messages received in sequence S_s in the following claim.

Claim 6 *There exists an order of the 4 rounds $B_k(s)$ of sequence S_s (rounds $4s$ to $4s+3$), namely $(B_{\alpha+1}(s), B_{\alpha-1}(s), B_\alpha(s), B_{\alpha+2}(s))$*

for n even, and $(B_\alpha(s), B_{\alpha+1}(s), B_{\alpha+2}(s), B_{\alpha-1}(s))$ *for n odd, such that during the sequence* S_s*:*

- *each node* $s+1 \le i \le n-s$ *has received message* $i-s-1$*, and each node* $s-1 \le i \le n-s-2$ *has received message* $i+s+1$*,*
- *nodes* $s-3j-2$ *(resp.* $n-s+3j+1$*),* $0 \le 3j \le s-2$ *have received two messages from the right (resp. from the left)*
- *and the other nodes* $i \le s-1$ *(resp.* $i \ge n-s$*) have received one message from the right (resp. from the left).*

Proof. The first part follows from the fact that for node $s+1 \le i \le n-s-2$, the calls are those of A_k, and so in one of the 4 rounds, each node receives a new message from the left, namely message $i-s-1$ and in another round receives a new message from the right, namely message $i+s+1$ (note that by induction these messages arrived at the sender at the end of sequence S_{s-1}). The orders determined in the preceding paragraph enable node $s-3j-1$ (resp. $n-s+3j$) to send two messages to node $s-3j-2$ (resp. $n-s+3j+1$), and also ensure the arrival of a message in the other nodes of the left and right. Therefore, the second and third parts are proven. ⋄

Messages received at the end of phase 2 (end of sequence S_{p-1})

Recall that $n = 2p$ or $2p+1$ and in phase 2, we do $p-1$ sequences S_s, $1 \le s \le p-1$.

Claim 7 *Let* $n = 2p$*. At the end of phase 2 (after round* $4p-1$*),*

- *nodes* $p-1-3j$*,* $0 \le 3j \le p-1$ *have received messages* $0 \le i \le 2p-1-2j$
- *nodes* $p-2-3j$*,* $0 \le 3j \le p-2$ *and* $p-3-3j$*,* $0 \le 3j \le p-3$ *have received messages* $0 \le i \le 2p-2-2j$
- *nodes* $p+3j$*,* $0 \le 3j \le p-1$ *have received messages* $2j \le i \le 2p-1$
- *nodes* $p+1+3j$*,* $0 \le 3j \le p-2$ *and* $p+2+3j$*,* $0 \le 3j \le p-3$ *have received messages* $2j+1 \le i \le 2p-1$

Let $n = 2p+1$*. At the end of phase 2 (after round* $4p-1$*),*

- *node p has received all the messages*
- *nodes* $p-1-3j$*,* $0 \le 3j \le p-1$ *have received messages* $0 \le i \le 2p-1-2j$
- *nodes* $p-2-3j$*,* $0 \le 3j \le p-2$ *and* $p-3-3j$*,* $0 \le 3j \le p-3$ *have received messages* $0 \le i \le 2p-2-2j$
- *nodes* $p+1+3j$*,* $0 \le 3j \le p-1$ *have received messages* $2j+1 \le i \le 2p$
- *nodes* $p+2+3j$*,* $0 \le 3j \le p-2$ *and* $p+3+3j$*,* $0 \le 3j \le p-3$ *have received messages* $2j+2 \le i \le 2p$

Proof. By claim 6, at the end of sequence S_{p-1}, any node i has received the messages of the nodes at distance $\le p$. Therefore, node $i \le p$ (resp. $i \ge n-1-p$) has received all the messages from the left (resp. right). In particular, node p has received all the messages and, when $n = 2p$, node $p-1$ has also received all the messages. Furthermore, node $i \le p$ (resp. $i \ge n-1-p$) has received more than p messages from the right (resp. left) since it has received in some sequences two messages. For the precise analysis we distinguish two cases according the parity of n.

Let $n = 2p$. As noted above, nodes $p-1$ and p have received all the messages. Node $p-2$ has received all the messages $0 \le i \le 2p-2$. But node $p-3$ has also received all the messages $0 \le i \le 2p-2$. Indeed in S_{p-1} it has received two messages namely $2p-3$ and $2p-2$. More generally, node $p-1-3j$ has received from the right two messages during the j sequences $S_{p-2-3k}, 0 \le k \le j-1$ and so it has received at the end of phase 2 from the right altogether $p+j$ messages, namely all the messages between $p-3j$ and $2p-1-2j$.

Node $p-2-3j$ has received two messages during the j sequences $S_{p-3-3k}, 0 \le k \le j-1$ and so has received at the end of phase 2 from the right all the messages between $p-1-3j$ and $2p-2-2j$. Node $p-3-3j$ has received two messages during the $j+1$ sequences $S_{p-1-3k}, 0 \le k \le j$ and so has received at the end of phase 2 from the right all the messages between $p-2-3j$ and $2p-2-2j$.

The proof for the other side is similar. Node $p+1$ has received the messages $1 \le i \le 2p-1$ at the end of phase 2. Node $p+2$ has also received the messages $1 \le i \le 2p-1$. Indeed in S_{p-1} it has received 2 messages namely 2 and 1. More generally node $p+3j$ has received from the lef, two messages during the j sequences $S_{p-2-3k}, 0 \le k \le j-1$ and so it has received at the end of phase 2 from the left, $p+j$ messages that are all the messages between $2j$ and $p+3j-1$. Node $p+3j+1$ has received two messages during the j sequences $S_{p-3-3k}, 0 \le k \le j-1$ and so it has received at the end of phase 2 from the left, $p+j$ messages that are all the messages between $2j+1$ and $p+3j$. Node $p+3j+2$ has received two messages during the $j+1$ sequences $S_{p-1-3k}, 0 \le k \le j$ and so it has received at the end of phase 2 from the left, $p+1+j$ messages that are all the messages between $2j+1$ and $p+3j+1$.

For $n = 2p+1$, the proof is similar as that for the case n even. ⋄

3.3 Phase 3

At the end of phase 2, the nodes in the left part $0 \le i \le p-1$ still have to receive some messages of large nodes and in particular, we have to move message $n-1$ up to the node 0 while the right part, the node $p+1 \le i \le n-1$ still have to receive some messages from small nodes and in particular, we have to move message 0 up to the node $n-1$. These moves can be completed independently since there will be no interferences between the calls in the left part and those in right part (except for the two first rounds in the case n odd). We have already completed $3+4(p-1) = 4p-1$ rounds in phases 1 and 2. So to complete the protocol in the optimal time, we should do phase 3 in $3n-4-4p$ rounds or namely, when $n = 2p$, in $2p-4$ rounds, and when $n = 2p+1$, in $2p-1$ rounds.

Claim 8 *We can construct* $3n-4-4p$ *rounds (phase 3) to complete the protocol in optimal time.*

Proof. The readers can again follow the proof on the tables given for $n = 12, 13, 14, 15$.

Let $n = 2p$. We first do the following two rounds. The first round contains the non-interfering calls $(p-1-3j, p-2-3j)$ and $(p+3j, p+1+3j)$ for $0 \le 3j \le p-2$, and the second round the calls $(p-2-3j, p-3-3j)$ and $(p+1+3j, p+2+3j)$ for $0 \le 3j \le p-3$. According to claim 7, after these two rounds, nodes $p-3$, $p-2$, $p+1$, and $p+2$ have received all the messages and nodes $p-4-3j, p-5-3j$, and $p-6-3j$ (resp. $p+3+3j, p+4+3j$, and $p+5+3j$) have received all messages $0 \le i \le 2p-3-2j$ (resp. $2j+2 \le i \le n-1$) for valid j.

The only remaining task is to now push the messages $n-2$ and $n-1$ (resp. 1 and 0) to the left (resp. right) via $p-3$ sequences T_k $(0 \leq k \leq p-4)$. Each T_k consists of two identical rounds each containing the calls $(p-3-k-3j, p-4-k-3j)$ and $(p+2+k+3j, p+3+k+3j)$ for $0 \leq 3j \leq p-4-k$. At the end of these sequences, each node has received all the messages. Altogether we have completed the protocol in $2+2\times(p-3) = 2p-4$ rounds.

Let $n = 2p+1$. We first do the following three rounds (the first two rounds enable us to separate the left and right parts). The first round contains the calls $(p-3j, p-1-3j)$, $0 \leq 3j \leq p-1$ and $(p+1+3j, p+2+3j)$ $0 \leq 3j \leq p-2$. The second round contains the calls $(p-1-3j, p-2-3j)$, $0 \leq 3j \leq p-2$, and $(p+3j, p+1+3j)$, $0 \leq 3j \leq p-1$. The third round contains the calls $(p-1-3j, p-2-3j)$ and $(p+1+3j, p+2+3j)$ for $0 \leq 3j \leq p-2$. According to claim 7 after these three rounds, nodes $p-1$, $p-2$, $p+1$, and $p+2$ have received all the messages. Nodes $p-3-3j$, $p-4-3j$, and $p-5-3j$ (resp. $p+3+3j$, $p+4+3j$, and $p+5+3j$) have received all messages $0 \leq i \leq 2p-2-2j$ (resp. $2j+2 \leq i \leq n-1$). Then we end the protocol just as in the case n even with the $p-2$ sequences T'_k, $0 \leq k \leq p-3$. Here, T'_k consists of two identical rounds each containing the calls $(p-2-k-3j, p-3-k-3j)$ and $(p+2+k+3j, p+2+k+3j)$ for $0 \leq 3j \leq p-3-k$. At the end of these sequences, each node has received all the messages. We have now completed the protocol in $3+2\times(p-2) = 2p-1$ rounds. ⋄

In summary, we have given a protocol in three phases which completes the gossiping for $n > 3$ in the optimal number of rounds $3n-5$ as given in Theorem 1.

4. Conclusion

In this article, we determine the exact minimum gossiping time in the chain network with n node using hypothetical unit length messages and neighboring interference. One can also try to determine the exact gossiping time for other simple topologies like grids. Perhaps one can use our tools for chains to improve the bounds for trees given in Ref. [6]. It will also be interesting to consider stronger interferences (a sending node prevents nodes at distance d_I to receive messages).

References

[1] Bermond, J.C., Correa, R. and Yu, M.L.: Optimal gathering protocols on paths under interference constraints, *Discrete Mathematics*, Vol.309, pp.5574–5587 (2009).

[2] Bermond, J.-C., Galtier, J., Klasing, R., Morales, N. and Pérennes, S.: Hardness and approximation of gathering in static radio networks, *Parallel Processing Letters*, Vol.16, No.2, pp.165–183 (2006).

[3] Bermond, J.C., Kodate, T. and Yu, M.L.: Gossiping with interferences in radio ring networks, available from ⟨https://hal.inria.fr/hal-02424099⟩ (preprint).

[4] Bonifaci, V., Klasing, R., Korteweg, P., Stougie, L. and Marchetti-Spaccamela, A.: Data Gathering in Wireless Networks chapter in Koster, A. and Munoz, X. (Eds.), *Graphs and Algorithms in Communication Networks*, pp.357–377. Springer Monograph (2009).

[5] Gąsieniec, L.: On efficient gossiping in radio networks, *Proc. Int. Conference on Theoretical Computer Science Colloquium on Structural Information and Communication Complexity, SIROCCO 2009*, Lectures Notes in Computer science, Vol.5869, pp.2–14, Springer Verlag (2010).

[6] Gąsieniec, L. and Potapov, I.: Gossiping with unit messages in known radio networks, *Proc. 2nd IFIP Int. Conference on Theoretical Computer Science*, pp.193–205 (2002).

[7] Klasing, R., Morales, N. and Pérennes, S.: On the complexity of bandwidth allocation in radio networks, *Theoretical Computer Science*, Vol.406, No.3, pp.225–239 (2008).

[8] Manne, F. and Xin, Q.: Optimal gossiping with unit size in known topology radio networks, *Proc. Workshop on Combinatorial and Algorithmic Aspects of Networking, CAAN 2006*, Lectures Notes in Computer science, Vol.4235, pp.125–134, Springer Verlag (2006).

Journal of Information Processing Vol.28 889–902 (Dec. 2020)

Appendix

Table A·1 $n = 12$.

round	s	calls	nodes											
			0	1	2	3	4	5	6	7	8	9	10	11
Phase1														
1	0	R_1	1		1	4		4	7		7	10		10
2		R_2	×	2		2	5		5	8		8	11	
3		R_3		0	3		3	6		6	9		9	×
Phase2														
4	1	A_0	×	×	4			3	8			7	×	×
5		A_1	×		0	5			4	9			8	×
6		A_2	2			1	6			5	10			9
7		A_3	×	3			2	7			6	11		×
8	2	B_3	×	4			1	8			5	×		8
9		B_1	3		×	6			3	10			7	×
10		B_2	4			0	7			4	11			7
11		B_0	×	×	5			2	9			6	×	×
12	3	B_0	×	×	6			1	10			5	×	×
13		B_2	×	5		×	8			3	×		6	×
14		B_3	×	6			0	9			4	×		6
15		B_1	5		×	7			2	11			5	×
16	4	B_1	6		×	8			1	×		4	×	×
17		B_3	×	×	7		×	10			3	×		5
18		B_0	×	×	8			0	11			3	×	×
19		B_2	×	7		×	9			2	×		4	×
20	5	B_2	×	8		×	10			1	×		3	×
21		B_0	7		×	9		×	×		2	×		4
22		B_1	8		×	10			0	×		2	×	×
23		B_3	×	×	9		×	11			1	×		3
Phase3														
24			×	9		×	11			0	×		2	×
25			9		×	11		×	×		0	×		2
26			×	×	10		×	×	×	×		1	×	×
27			×	×	11		×	×	×	×		0	×	×
28			×	10		×	×	×	×	×	×		1	×
29			×	11		×	×	×	×	×	×		0	×
30			10		×	×	×	×	×	×	×	×		1
31			11		×	×	×	×	×	×	×	×		0

Table A·2 $n = 13$.

round	s	calls	nodes												
			0	1	2	3	4	5	6	7	8	9	10	11	12
Phase1															
1	0	R_1	1		1	4		4	7		7	10		10	×
2		R_2	×	2		2	5		5	8		8	11		11
3		R_3		0	3		3	6		6	9		9	12	
Phase2															
4	1	A_0	×	×	4			3	8			7	12		×
5		A_1	×		0	5			4	9			8	×	×
6		A_2	2			1	6			5	10			9	×
7		A_3	×	3			2	7			6	11			10
8	2	B_2	3			0	7			4	11			8	×
9		B_3	×	4			1	8			5	12			9
10		B_0	×	×	5			2	9			6	×		8
11		B_1	4		×	6			3	10			7	×	×
12	3	B_3	×	5			0	9			4	×		7	×
13		B_0	×	×	6			1	10			5	×		7
14		B_1	5		×	7			2	11			6	×	×
15		B_2	×	6		×	8			3	12			6	×
16	4	B_0	×	×	7			0	11			4	×		6
17		B_1	6		×	8			1	12			5	×	×
18		B_2	×	7		×	9			2	×		4	×	×
19		B_3	×	×	8		×	10			3	×		5	×
20	5	B_1	7		×	9			0	×		3	×		5
21		B_2	×	8		×	10			1	×		3	×	×
22		B_3	×	×	9		×	11			2	×		4	×
23		B_0	8		×	10		×	12			2	×		4
Phase3															
24			×	×	10		×	12			1	×		3	×
25			×	9		×	11			0	×		2	×	×
26			×	10		×	12		×		0	×		2	×
27			9		×	11		×	×	×		1	×		3
28			10		×	12		×	×	×		0	×		2
29			×	×	11		×	×	×	×	×		1	×	×
30			×	×	12		×	×	×	×	×		0	×	×
31			×	11		×	×	×	×	×	×	×		1	×
32			×	12		×	×	×	×	×	×	×		0	×
33			11		×	×	×	×	×	×	×	×	×		1
34			12		×	×	×	×	×	×	×	×	×		0

Journal of Information Processing Vol.28 889–902 (Dec. 2020)

Table A·3 $n = 14$.

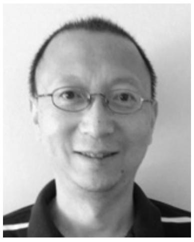
round	s	calls	nodes													
			0	1	2	3	4	5	6	7	8	9	10	11	12	13
Phase1																
1	0	R_1	1		1	4		4	7		7	10		10	13	
2		R_2	×	2		2	5		5	8		8	11		11	×
3		R_3		0	3		3	6		6	9		9	12		12
Phase2																
4	1	A_0	×	×	4			3	8			7	12			11
5		A_1	×		0	5			4	9			8	13		×
6		A_2	2			1	6			5	10			9	×	×
7		A_3	×	3			2	7			6	11			10	×
8	2	B_3	×	4			1	8			5	12			9	×
9		B_1	3		×	6			3	10			7	×		10
10		B_2	4			0	7			4	11			8	×	×
11		B_0	×	×	5			2	9			6	13			9
12	3	B_0	×	×	6			1	10			5	×		8	×
13		B_2	×	5		×	8			3	12			7	×	×
14		B_3	×	6			0	9			4	13			7	×
15		B_1	5		×	7			2	11			6	×		8
16	4	B_1	6		×	8			1	12			5	×		7
17		B_3	×	×	7		×	10			3	×		6	×	×
18		B_0	×	×	8			0	11			4	×		6	×
19		B_2	×	7		×	9			2	13			5	×	×
20	5	B_2	×	8		×	10			1	×		4	×		6
21		B_0	7		×	9		×	12			3	×		5	×
22		B_1	8		×	10			0	13			3	×		5
23		B_3	×	×	9		×	11			2	×		4	×	×
24	6	B_3	×	×	10		×	12			1	×		3	×	×
25		B_1	×	9		×	11		×	×		2	×		4	×
26		B_2	×	10		×	12			0	×		2	×		4
27		B_0	9		×	11		×	13			1	×		3	×
Phase3																
28			×	×	11		×	13			0	×		2	×	×
29			×	11		×	13		×	×		0	×		2	×
30			10		×	12		×	×	×	×		1	×		3
31			11		×	13		×	×	×	×		0	×		2
32			×	×	12		×	×	×	×	×	×		1	×	×
33			×	×	13		×	×	×	×	×	×		0	×	×
34			×	12		×	×	×	×	×	×	×	×		1	×
35			×	13		×	×	×	×	×	×	×	×		0	×
36			12		×	×	×	×	×	×	×	×	×	×		1
37			13		×	×	×	×	×	×	×	×	×	×		0

Table A·4 $n = 15$.

round	s	calls	nodes														
			0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Phase1																	
1	0	R_1	1		1	4		4	7		7	10		10	13		13
2		R_2	×	2		2	5		5	8		8	11		11	14	
3		R_3		0	3		3	6		6	9		9	12		12	×
Phase2																	
4	1	A_0	×	×	4			3	8			7	12			11	×
5		A_1	×		0	5			4	9			8	13			12
6		A_2	2			1	6			5	10			9	14		×
7		A_3	×	3			2	7			6	11			10	×	×
8	2	B_2	3			0	7			4	11			8	×		11
9		B_3	×	4			1	8			5	12			9	×	×
10		B_0	×	×	5			2	9			6	13			10	×
11		B_1	4		×	6			3	10			7	14			10
12	3	B_3	×	5			0	9			4	13			8	×	×
13		B_0	×	×	6			1	10			5	14			9	×
14		B_1	5		×	7			2	11			6	×		8	×
15		B_2	×	6		×	8			3	12			7	×		9
16	4	B_0	×	×	7			0	11			4	×		7	×	×
17		B_1	6		×	8			1	12			5	×		7	×
18		B_2	×	7		×	9			2	13			6	×		8
19		B_3	×	×	8		×	10			3	14			6	×	×
20	5	B_1	7		×	9			0	13			4	×		6	×
21		B_2	×	8		×	10			1	14			5	×		7
22		B_3	×	×	9		×	11			2	×		4	×		6
23		B_0	8		×	10		×	12			3	×		5	×	×
24	6	B_2	×	9		×	11			0	×		3	×		5	×
25		B_3	×	×	10		×	12			1	×		3	×		5
26		B_0	9		×	11		×	13			2	×		4	×	×
27		B_1	×	10		×	12		×	14			2	×		4	×
Phase3																	
28			10		×	12		×	14			1	×		3	×	×
29			×	×	11		×	13			0	×		2	×		4
30			×	×	12		×	14		×		0	×		2	×	×
31			×	11		×	13		×	×	×		1	×		3	×
32			×	12		×	14		×	×	×		0	×		2	×
33			11		×	13		×	×	×	×	×		1	×		3
34			12		×	14		×	×	×	×	×		0	×		2
35			×	×	13		×	×	×	×	×	×	×		1	×	×
36			×	×	14		×	×	×	×	×	×	×		0	×	×
37			×	13		×	×	×	×	×	×	×	×	×		1	×
38			×	14		×	×	×	×	×	×	×	×	×		0	×
39			13		×	×	×	×	×	×	×	×	×	×	×		1
40			14		×	×	×	×	×	×	×	×	×	×	×		0

Jean-Claude Bermond was born in 1945. He received his "Doctorat d'État" from university Paris Sud (France) in 1975. He is currently Research Director (Emeritus) at CNRS in the project COATI (join project between the laboratory I3S of UCA (Université Côte d'Azur) and Inria). He has been actively involved in research in the areas of Discrete Mathematics, Interconnection Networks and Parallel and Distributed Computing with applications in Telecommunications. In these fields he has published more than 200 papers and supervised 50 PhDs. He is member of many editorial boards of Journals. He has also been chair of different laboratories. He won the "Grand Prix" EADS in Computer Science awarded by the French Academy of science and the prize for Innovation in Distributed Computing.

Takako Kodate was born in 1968. She received her M.S. from Tokyo Woman's Christian University in 1992 and Doctorat en Sciences from Université de Nice-Sophia Antipolis in 1996. She is currently a lecturer at Tokyo Woman's Christian University. Her research interest concerns communication problems in networks.



Joseph Yu was born in 1961. He received his Ph.D. from Simon Fraser University (Canada) in 1990. He is currently an associate professor in the department of Mathematics and Statistics at University of the Fraser Valley (Canada). His research interest is in the areas of Discrete Mathematics and Graph Theory in the network applications.